OAuth Working Group
Internet-Draft
Intended status: Informational
Expires: 29 July 2026

F. Morin
FXCO Ltd.

25 January 2026

OAuth 2.1 Government Content Access Control
draft-fx-oauth-government-content-access-control-02

Abstract

This document defines an OAuth 2.1 profile that enables a government authority to enforce age-based and content-based access restrictions for online services while preserving user privacy. The protocol allows relying parties to request government-defined regulatory scopes (such as pornography or social media access) and receive cryptographically verifiable eligibility decisions without disclosing user identity, exact age, or personally identifiable information. The profile constrains OAuth features to prevent abuse, cross-service correlation, and unauthorized token issuance.

Foreword

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://fxmorin.github.io/government-content-access-control.

Status information for this document may be found at https://datatracker.ietf.org/doc/draft-fx-oauth-government-content-access-control/.

Discussion of this document takes place on the Web Authorization Protocol Working Group mailing list (mailto:oauth@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/oauth/.

Subscribe at https://www.ietf.org/mailman/listinfo/oauth/.

Source for this draft and an issue tracker can be found at https://github.com/FxMorin/government-content-access-control

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119] & [RFC8174].

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Governments increasingly require online services to restrict access to certain categories of content based on the age or legal status of users. Existing approaches frequently rely on disclosure of personal data, third-party identity providers, or proprietary mechanisms that enable tracking across services.

This document specifies OAuth 2.1 Government Content Access Control (GCAC), an OAuth 2.1 [I-D.ietf-oauth-v2-1] profile in which a government-operated authorization server evaluates user eligibility for regulated content categories and issues privacy-preserving attestations to relying parties. GCAC is designed to answer narrowly scoped regulatory questions (e.g., whether a user may access a category of content) while minimizing data disclosure and preventing correlation across services.

GCAC is not an identity system and MUST NOT be used for authentication, user login, or personalization.

2. Definitions

The following terms are used:

Government Content Control Authority (GCCA):

- A government-operated OAuth Authorization Server responsible for identity verification, age evaluation, and policy enforcement.

Relying Party (RP):

- An OAuth client requesting eligibility decisions for regulated content.

Scope:

- A government-defined content access category (e.g., pornography, social_media, gambling, alcohol, firearms, vpn, proxy).

Person Key:

- A government-internal, pseudonymous identifier derived from a national identity record and never exposed outside the GCCA.

Site Token:

- An RP-scoped, non-reversible token representing a government eligibility attestation.

3. Goals and Non-Goals

3.1. Goals

The goals of this specification are:

Enable government-enforced content access controls

Support multiple regulatory scopes defined by government policy

Prevent disclosure of identity, date of birth, or exact age

Prevent cross-RP correlation of users

Leverage existing OAuth 2.1 security mechanisms

3.2. Non-Goals

This specification explicitly does not attempt to:

Provide user authentication or login services to RPs

Expose personal attributes or identity claims (e.g., OIDC claims)

Enable cross-service user identification

Replace digital identity or credential systems

4. Architecture Overview

GCCA uses the OAuth 2.1 Authorization Code flow with mandatory security extensions and additional semantic constraints.

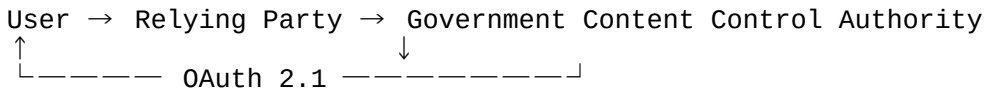

Figure 1: GCCA Code Flow

The GCCA operates as a constrained OAuth Authorization Server, and the RP operates as a confidential OAuth client.

5. Scope Model

5.1. Government-Defined Scopes

Scopes represent legally regulated content categories. Examples include:

- pornography
- social_media
- gambling
- alcohol
- firearms
- vpn
- proxy

Each scope:

- MUST be defined and governed by the GCCA
- MUST correspond to a legal or regulatory access rule
- MUST prevent scope definitions or combinations thereof that would allow an RP to infer a user's exact age or approximate age range through multiple eligibility queries. Scopes MUST be coarse-grained and legally motivated, and MUST NOT be parameterized by numeric age values.

5.2. Scope Evaluation Semantics

For each requested scope, the GCCA determines whether the user satisfies the applicable legal requirement. The RP receives only a boolean eligibility result per scope.

6. Client Registration

Relying parties MUST register with the GCCA prior to using GCAC.

During registration, the RP MUST provide:

- Legal entity identification
- Intended use and justification for requested scopes
- One or more redirect URIs
- A client authentication method (mutual TLS [RFC8705] or private_key_jwt)

The GCCA MAY restrict which scopes an RP is authorized to request.

7. Protocol Flow

7.1. Authorization Request

The RP initiates an OAuth authorization request:

```
GET /authorize
?response_type=code
&client_id=client_id
&redirect_uri=registered_uri
&scope=pornography+social_media
&state=random_nonce
&code_challenge=pkce_value
```

Figure 2: Example: GCCA request

The GCCA MUST reject requests that include unregistered redirect URIs or unauthorized scopes.

7.2. User Authentication and Evaluation

The GCCA authenticates the user using government-controlled mechanisms and evaluates eligibility for each requested scope.

7.3. Authorization Response

Upon successful evaluation, the GCCA redirects the user back to the RP with an authorization code.

7.4. Token Request and Response

The RP exchanges the authorization code at the token endpoint using client authentication and PKCE [RFC7636].

The GCCA responds with a site-scoped eligibility attestation as a standard OAuth 2.1 token response, including additional custom parameters:

```
HTTP/1.1 200 OK
Content-Type: application/json
Cache-Control: no-store

{
  "access_token": "opaque_site_token_string",
  "token_type": "Bearer",
  "expires_in": 3600,
  "scope_results": {
    "pornography": true,
    "social_media": false
  }
}
```

Figure 3: Example: GCCA Response

The "access_token" field contains the site token, which is an RP-scoped, non-reversible eligibility attestation.

8. Token Derivation

The GCCA MUST derive an internal person key using HMAC-SHA256 with a globally unique master secret key (master_secret_key) and a stable, unique national identifier for the person (national_person_id):

```
person_key = HMAC_SHA256(master_secret_key, national_person_id)
```

Figure 4: Person Key Pseudocode

The site token MUST be derived deterministically using HMAC-SHA256 from the person key (person_key) and the client identifier (client_id):

```
site_token = HMAC_SHA256(person_key, client_id)
```

Figure 5: Site Token Pseudocode

The person key and national identifiers MUST NOT be exposed outside the GCCA.

9. Re-Verification

Relying parties MUST provide a user-accessible mechanism to re-initiate the GCAC flow. Re-verification MUST follow the same protocol as the initial authorization.

10. Security Considerations

GCAC relies on OAuth 2.1 security best practices, including authorization code flow, PKCE [RFC7636], redirect URI allowlists, and strong client authentication. General security considerations for OAuth 2.0 [RFC6819] also apply. To further protect the integrity and confidentiality of authorization requests, RPs SHOULD use JWT-Secured Authorization Request (JAR) [RFC9101]. Leaked client identifiers alone do not enable token issuance. Tokens are RP-scoped and non-transferable, preventing cross-service correlation and replay attacks.

11. IANA Considerations

This document has no IANA actions.

12. References

12.1. Normative References

[I-D.ietf-oauth-v2-1]
Hardt, D., Parecki, A., and T. Lodderstedt, "The OAuth 2.1 Authorization Framework", Work in Progress, Internet-Draft, draft-ietf-oauth-v2-1-11, 20 January 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-oauth-v2-1>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC7636] Sakimura, N., Ed., Bradley, J., and N. Agarwal, "Proof Key for Code Exchange by OAuth Public Clients", RFC 7636, DOI 10.17487/RFC7636, September 2015, <https://www.rfc-editor.org/info/rfc7636>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8705] Campbell, B., Bradley, J., Sakimura, N., and T. Lodderstedt, "OAuth 2.0 Mutual-TLS Client Authentication and Certificate-Bound Access Tokens", RFC 8705, DOI 10.17487/RFC8705, February 2020, <https://www.rfc-editor.org/info/rfc8705>.

12.2. Informative References

[RFC6819] Lodderstedt, T., Ed., McGloin, M., and P. Hunt, "OAuth 2.0 Threat Model and Security Considerations", RFC 6819, DOI 10.17487/RFC6819, January 2013, <https://www.rfc-editor.org/info/rfc6819>.

[RFC9101] Sakimura, N., Bradley, J., and M. Jones, "The OAuth 2.0 Authorization Framework: JWT-Secured Authorization Request (JAR)", RFC 9101, DOI 10.17487/RFC9101, August 2021, <https://www.rfc-editor.org/info/rfc9101>.

Acknowledgments

The author would like to acknowledge ongoing discussions within the OAuth and digital privacy communities that informed the design principles of this specification.

Author’s Address

Francois-Xavier Morin
FXCO Ltd.
Canada
Email: fx@fxco.ca
URI: https://fxco.ca/